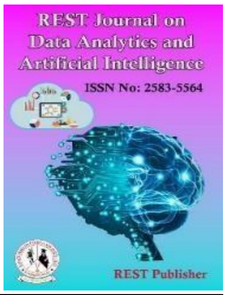
REST Journal on Data Analytics and Artificial Intelligence
Vol: 3(3), September 2024
REST Publisher; ISSN: 2583-5564
Website: http: //restpublisher.com/journals/jdaai/
DOI: https://doi.org/10.46632/jdaai/3/3/1



Evolution of Structured Deep Visual Models in Robot Manipulation

***[1]Karamala Naveen, [2]Roopa R, [3]T Harikala, [4]P. Anjaiah**
[1]Vemu Institute of Technology, Pakala, Kerala, India.
[2]Madanapalle Institute of Technology & Science, Madanapalle, India.
[3]Annamacharya University, Rajampet. India.
[4]Mallareddy University, Medchal (Dist), Hyderabad. India.
**Corresponding author: karamalanaveen@gmail.com*

Abstract: *This paper explores the intersection of deep learning and robotics, focusing on the development and implementation of structured deep visual models for improving the perception and control of robotic systems in manipulation tasks. The integration of convolution neural networks (CNNs), recurrent neural networks (RNNs), and attention mechanisms plays a pivotal role in enabling robots to efficiently interpret visual data and make informed decisions in complex and dynamic environments. The presented research contributes to the ongoing efforts in bridging the gap between perception and action in robotics, paving the way for more robust and intelligent manipulation capabilities in diverse and challenging environments. The insights gained from this study offer valuable guidance for researchers, engineers, and practitioners working on the forefront of advancing robotic systems through the integration of deep learning techniques. Structured deep visual models contribute to improving a robot's ability to perceive and understand its environment. This includes advancements in object recognition, pose estimation, and scene understanding, allowing robots to interact with their surroundings more intelligently. The integration of deep learning in robot manipulation empowers robots to operate with greater autonomy. Structured visual models enable robots to make informed decisions based on visual data, reducing the need for explicit programming and enhancing adaptability in dynamic environments. The acceptability of alternatives is gauged by comparing them to the average response. Utilizing the EDAS method, this assessment determines the most advantageous answer by considering both the average evaluation and its deviation from the mean solution. According to the analysis, EDAS favors solutions closer to the ideal solution, while penalizing those with negative deviations, indicating a preference for options that closely align with the ideal. From the result Graph Neural Networks (GNNs) is got the first rank where as Generative Adversarial Networks (GANs) is having the lowest rank.*

1. INTRODUCTION

Deep learning plays a pivotal role in advancing robotics, employing various model designs and training methodologies. Convolutional neural networks (CNNs) are frequently employed for tasks involving perception, while recurrent networks facilitate sequential decision-making processes. Reinforcement learning is instrumental in refining robotic actions through iterative trial and error. Additionally, transfer learning facilitates the transfer of knowledge across different domains. The integration of these strategies with simulation-based training is crucial for enhancing the robustness and flexibility of robotic systems. [1] In the realm of real-time perception, an innovative deep learning system has been developed for visual servo control and grip detection in autonomous manipulation robots. Leveraging CNNs, this system allows robots to swiftly adapt their movements by processing visual input, thereby ensuring precise grasp detection and manipulation. This advancement enhances the versatility and responsiveness of unmanned robotic systems operating in intricate environments. [2] Working with flexible objects holds significance across diverse fields like medical procedures, industrial production, and everyday household robotics. Managing deformable materials such as ropes, cables, and hoses poses significant challenges for robots due to the absence of precise analytical models and the multitude of possible configurations. Additionally, teaching robotic manipulation solely through visual inputs and direct physical interaction demands extensive training and may struggle to adapt to varying tasks. [3] Mobile service robots are capable of executing various practical duties, including guiding tours, delivering products, cleaning, monitoring, and assisting in healthcare. Mobile manipulators excel in tasks like object localization and pick-and-place operations. To seamlessly integrate into human-centric environments, these robots must have a small footprint and the ability to interact physically with their surroundings. However, conventional wheeled robots often have bulky bases to prevent tipping, which compromises their speed and agility. The idea of autonomous wheeled robots addresses these challenges effectively. Illustrated by the Ball-bot, which stays dynamically stable on a single spherical wheel, this design minimizes the possibility of tipping over while enabling the support of taller structures for enhanced interaction with humans and reducing the overall weight of the robot. The Ball-bot prototype utilizes an existing self-balancing two-wheeled scooter known as the Ninebot Mini Pro. This scooter incorporates a motion mechanism capable of detecting changes in tilt angle and subsequently adjusting its forward propulsion, thereby aiding the machine in maintaining balance and mobility. The Sawyer robot employs a technique called soft Q learning, a form of maximum entropy reinforcement learning, to assemble Lego pieces. The training of a policy from scratch takes less than two hours, yielding a policy that demonstrates remarkable resilience to disturbances. The concept of compositionality suggests that multiple policies can be amalgamated to create a new policy capable of addressing all tasks assigned to its component policies. This trait is advantageous as it facilitates the reutilization and swift initiation of policies, enabling the learning of complex compound skills based on previously acquired building blocks. Todorov has proposed a similar notion, investigating the integration of independent rewards through soft maximization. However, such a form of composition typically addresses each constituent task separately, resulting in a disjointed approach. Conversely, our approach to composition combines tasks, typically offering greater utility (e.g., simultaneous navigation towards a target while avoiding obstacles) [5]. Visual Servo Control of Cable-driven Soft Robotic Manipulator involves utilizing visual feedback to precisely govern the movements of a soft robotic manipulator propelled by cables. By employing computer vision algorithms, this approach facilitates real-time adjustments, enabling the soft robot to respond dynamically to its environment. Such methodology enhances flexibility and precision across various applications, including delicate object manipulation and navigation through constrained spaces [6] Deep reinforcement learning (DRL) for robotic manipulation control revolves around training neural networks to iteratively make decisions within intricate scenarios. Through iterative experimentation, the system discerns optimal actions for tasks like grasping and manipulating objects. DRL's inherent adaptability and ability to navigate uncertainty render it a potent strategy for augmenting the autonomy and adaptability of robotic manipulation systems [7] Deep effect trajectory prediction in robot manipulation employs deep learning techniques to anticipate the trajectory of an object undergoing manipulation by a robot. This method utilizes neural networks to grasp intricate relationships, enabling accurate forecasts of the impact of the robot's maneuvers on the object's path. These predictions enhance the robot's ability to strategize and execute meticulous manipulation tasks effectively. [9] Cognitive radio (CR) emerges as the fitting technological remedy for addressing radio resource scarcity and the presence of shared channels. Deploying CR systems necessitates the implementation of effective sensing procedures to continually monitor channel conditions. Yet, achieving accurate insights into resource status relies heavily on the collaboration of multiple sensing devices. [10] Humans often rely on recognizable images or landmarks to navigate, while traditional robotic navigation methods require precise mapping, localization, and planning, making them vulnerable to slight environmental changes. PoliNet, however, is a deep visual model predictive control-policy learning system designed to facilitate visual navigation and prevent collisions with unseen obstacles along the path. By leveraging visual trajectory and 360-degree images from the robot's current perspective, PoliNet generates velocity commands for a planning horizon of N steps, optimizing trajectories for obstacle avoidance.

2. MATERIALS AND METHODS

2.1. Alternative parameters: Convolution Neural Networks (CNNs), Graph Neural Networks (GNNs), Recurrent Neural Networks (RNNs), Recurrent Neural Networks (RNNs), Generative Adversarial Networks (GANs), Hierarchical Reinforcement Learning (HRL), Attention Mechanism-based Networks

2.2. Evaluation parameters: Accuracy (%), Computational Cost (FLOPs), Number of Parameters, Memory Footprint (MB)

2.3. Convolution Neural Networks (CNNs): Convolution Neural Networks (CNNs) are deep learning models designed for tasks like image recognition and classification. They employ specialized layers called convolution layers to automatically learn hierarchical patterns from input data. These layers use filters to convolve across the input image, extracting features such as edges and textures. Through repeated application of convolution and pooling layers, CNNs progressively learn more complex features, enabling them to recognize objects with remarkable accuracy and robustness.

2.4. Graph Neural Networks (GNNs): Graph Neural Networks (GNNs) are a class of neural networks tailored for analyzing and learning from graph-structured data. Unlike traditional neural networks, GNNs operate directly on graphs, leveraging node and edge relationships. They iteratively update node representations by aggregating information from neighboring nodes, enabling them to capture complex graph structures and perform tasks like node classification, link prediction, and graph classification. GNNs have found applications in social network analysis, recommendation systems, drug discovery, and various other domains where data is naturally represented as graphs.

2.5. Generative Adversarial Networks (GANs): Generative Adversarial Networks (GANs) are a class of deep learning frameworks where two neural networks, the generator and the discriminator, compete against each other. The generator generates synthetic data, such as images, from random noise, while the discriminator tries to distinguish between real and fake data. Through adversarial training, GANs learn to produce increasingly realistic samples. GANs have gained widespread attention for their ability to generate high-quality images, and they find applications in image synthesis, data augmentation, and anomaly detection.

2.6. Hierarchical Reinforcement Learning (HRL): Hierarchical Reinforcement Learning (HRL) is a framework in reinforcement learning where agents learn and execute actions at multiple levels of abstraction. Instead of dealing with a flat action space, HRL organizes actions hierarchically, allowing for more efficient exploration and decision-making in complex environments. At higher levels, agents make decisions about broad goals or sub-tasks, while lower levels handle finer-grained actions to achieve these goals. HRL aims to improve sample efficiency, enhance generalization, and enable agents to tackle tasks with long time horizons or intricate structures more effectively.

2.7. Attention Mechanism-based Networks: Attention Mechanism-based Networks are a class of neural networks that dynamically weigh the importance of different parts of input data during processing. Inspired by human attention, these networks learn to focus on relevant information while disregarding irrelevant details. In tasks like machine translation and image captioning, attention mechanisms enable models to selectively attend to specific words or image regions, improving performance by allowing the model to focus on relevant context. These networks have revolutionized various fields by enhancing model interpretability and performance in tasks requiring context-aware processing.

2.8. Accuracy (%): Accuracy refers to the degree of conformity between a measurement and the true value of what is being measured. In various fields like statistics, science, and technology, accuracy is crucial for reliable results and decision-making. It reflects the absence of errors or bias in measurements or predictions. Achieving high accuracy often involves rigorous testing, calibration, and validation processes. In data analysis, accuracy measures how close a model's predictions are to the actual outcomes. It's a fundamental metric in assessing the quality and trustworthiness of information and methodologies employed in diverse disciplines.

2.9. Computational Cost (FLOPs): Computational cost, often quantified in terms of Floating Point Operations (FLOPs), denotes the amount of arithmetic operations performed by a computational process. It serves as a measure of the workload or complexity involved in executing algorithms or tasks. Higher FLOPs typically indicate more intensive computations, requiring more time and resources to complete. Understanding computational cost is vital for optimizing algorithms, selecting suitable hardware configurations, and managing computational resources efficiently, especially in fields like artificial intelligence, scientific computing, and computer graphics.

2.10. Number of Parameters: The number of parameters refers to the quantity of variables in a mathematical model, particularly prevalent in machine learning and neural networks. These parameters are the values that the model adjusts during the training process to minimize error and improve performance. A higher number of parameters often implies greater model complexity and capacity to learn intricate patterns from data. However, excessive parameters can lead to overfitting and increased computational requirements, necessitating a balance between model complexity and generalization ability.

2.11. Memory Footprint (MB): Memory footprint, typically measured in megabytes (MB), refers to the amount of memory space required to store and execute a program or process. It encompasses the total memory usage by an application, including code, data, and resources, while it is running. Understanding memory footprint is crucial for optimizing software performance and resource management, especially in environments with limited memory capacity such as embedded systems or mobile devices. Minimizing memory footprint helps enhance efficiency and enables smoother execution of applications across various computing platforms.

2.12. Method: The EDAS score serves as a tool for determining the energy requirements of a manufacturing facility, predominantly influenced by its proximity to recommended processing equipment. Although there exists a disparity between expert opinions and generated data regarding solar and geothermal energy, solar energy is favored by experts due to its environmental friendliness and widespread accessibility, ranking second in the Fuzzy AHP context. However, its adoption is hindered by high installation costs and subpar performance, as noted in reference [13]. The EDAS method is suggested for assessing energy sources in the stock category due to its superior accuracy and reduced mathematical complexity compared to other categorization approaches. It is widely recognized for its scalability and standard solution capabilities. Moreover, an enhanced version of the EDAS technique for supplier selection is proposed based on character replacement position. In the context of solid waste management, intuition derived from the EDAS approach recommends employing a fuzzy model to pinpoint suitable sites. Reference [14] demonstrates the utilization of EDAS for investigating the constraints of renewable energy production.

The EDAS method is applied in multiple-criteria group decision-making (MCGDM). Initially, it involves defining projects and employing the distance strategy, which is then expanded through EDAS. This methodology finds practical application and is influenced by the EDAS method. It offers a distinctive and environmentally friendly solution to the MCDM problem with inverse properties. The AVS is employed to prioritize options and assess them using the PDA and NDA EDAS methods. To address challenges related to Multiple Criteria Decision Making (MCDM), EDAS presents a distinctive solution, functioning as both a comprehensive system and framework. Based on existing literature, the extended EDAS model performs optimally when grounded in intuitive parametric difference measurements. Additionally, it serves as an empirical approach to sanitary waste disposal, aiding in the resolution of issues in evaluating initial waste disposal procedures for sanitation while ensuring result robustness for the proposed approach. Several recent approaches are compared to assess the accuracy of the findings. The EDAS strategy has been improved to incorporate the DHHFL framework for achieving carbon neutrality, which could lead Indian Smart Cities to significantly reduce their carbon emissions by 2050. EDAS relies solely on distance metrics, and its ranking algorithm is derived from the average of the Sweet and Nadir statistical components [19]. While EDAS stands out as one of the most widely used MCDM methods, it presents itself as a feasible alternative as well [20]. Particularly in supplier selection, it proves to be valuable, as evidenced by the "EDAS Supplier Selection Methodology." However, it is important to note that there is a scarcity of research in the current academic literature exploring MADM challenges using the EDAS approach. Consequently, employing EDAS in MADM presents an intriguing avenue for research, offering opportunities for evaluating and identifying prospects within a single-valued neutrosophic clean environment [21]. The EDAS methodology, also referred to as "estimation distance from the mean solution based," offers a fresh and effective approach to addressing stock-related challenges. Its efficacy is demonstrated through comparisons with various Multi-Criteria Decision Making (MCDM) techniques. Additionally, a fuzzy extension of EDAS is developed for provider selection, while a simplified version is crafted for choosing a dependable waste disposal site. The document also outlines several decision-making strategies employing neutrosophic units, all rooted in the principles of the EDAS methodology. In terms of order allocation, an EDAS-based mechanism is introduced, considering dealer evaluations and contextual factors. This method integrates stages from the EDAS approach and IT2FS mathematical tools to evaluate suppliers' compliance with environmental criteria.

3. RESULT AND DISCUSSION

TABLE 1. Evolution of structured deep visual models in robot manipulation

	Accuracy (%)	Computational Cost (FLOPs)	Number of Parameters	Memory Footprint (MB)
Convolutional Neural Networks (CNNs)	92.5	50	10	200
Graph Neural Networks (GNNs)	91.7	45	8	180
Recurrent Neural Networks (RNNs)	93.2	55	12	220
Generative Adversarial Networks (GANs)	90.8	60	15	240
Hierarchical Reinforcement Learning (HRL)	92	48	9	190
Attention Mechanism-based Networks	94.2	50	13	210
AVj	**92.40000**	**51.33333**	**11.16667**	**206.66667**

Table 1 shows compare above values. Accuracy: Attention Mechanism-based Networks have the highest accuracy at 94.2%, while GANs have the lowest at 90.8%. Computational Cost (FLOPs): GANs have the highest computational cost at 60 FLOPs, while GNNs have the lowest at 45 FLOPs. Number of Parameters: GANs have the highest number of parameters at 15, while GNNs and HRL have the lowest at 8 and 9 respectively. Memory Footprint: GANs have the highest memory footprint at 240 MB, while GNNs have the lowest at 180 MB. Attention Mechanism-based Networks exhibit the highest accuracy, but with a moderate computational cost and memory footprint. GNNs have relatively lower computational cost and memory footprint compared to others, while still maintaining a high accuracy level.

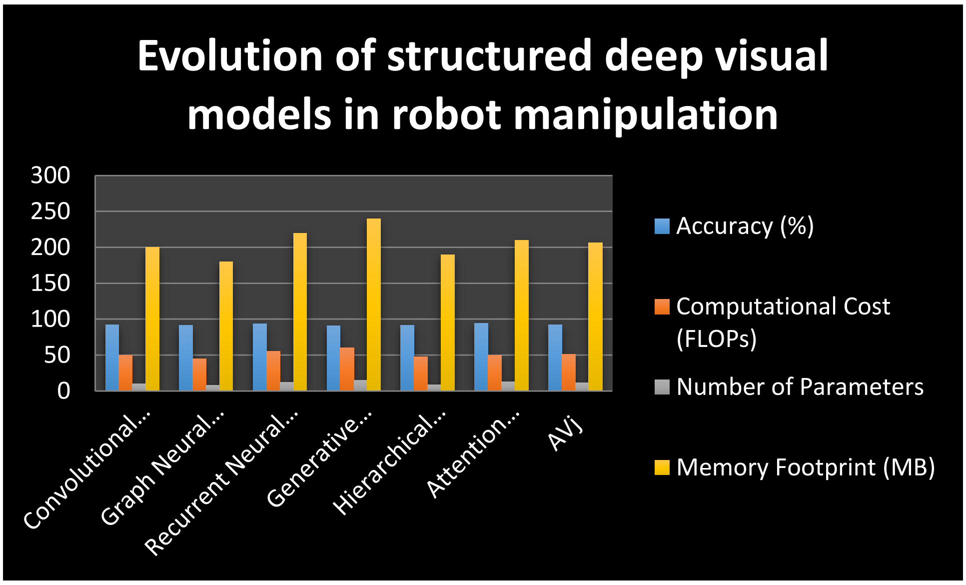


FIGURE 1. Evolution of structured deep visual models in robot manipulation

Figure 1illustrate graphical representation of Evolution of structured deep visual models in robot manipulation

TABLE 2. Positive Distance from average (PDA)

Positive Distance from Average (PDA)			
0.00	0.00	0.10	0.03
0.00	0.00	0.28	0.13
0.01	0.07	0.00	0.00
0.00	0.17	0.00	0.00
0.00	0.00	0.19	0.08
0.02	0.00	0.00	0.00

Table 2 shows the positive distance from the average Convolution Neural Networks (CNNs): The NDA values for CNNs across different categories are 0.00000 for the first category, 0.02597 for the second category, and 0.00000 for the third and fourth categories. This suggests that CNNs perform very close to the average in the second category, while they perform similarly to the average in the first and third categories, and exceptionally well (with 0 distance from the average) in the fourth category. Graph Neural Networks (GNNs): For GNNs, the NDA values are 0.00758 for the first category, 0.12338 for the second category, and 0.00000 for the third and fourth categories. This indicates that GNNs perform slightly better than average in the first category, significantly better than average in the second category, and on par with the average in the third and fourth categories. Recurrent Neural Networks (RNNs): The NDA values for RNNs are 0.00000 for the first and third categories, and 0.07463 for the second category, and 0.06452 for the fourth category. This implies that RNNs perform exactly at the average level in the first and third categories, slightly below average in the second category, and moderately above average in the fourth category. Generative Adversarial Networks (GANs): GANs exhibit NDA values of 0.01732 for the first category, 0.00000 for the second category, 0.34328 for the third category, and 0.16129 for the fourth category. This suggests that GANs perform slightly better than average in the first category, exactly at the average level in the second category, significantly below average in the third category, and moderately above average in the fourth category. Hierarchical Reinforcement Learning (HRL): For HRL, the NDA values are 0.00433 for the first category, 0.06494 for the second category, and 0.00000 for the third and fourth categories. This indicates that HRL performs slightly better than average in the first category, better than average in the second category, and at the average level in the third and fourth categories. Attention Mechanism-based Networks: The NDA values for attention mechanism-based networks are 0.00000 for the first category, 0.02597 for the second category, 0.16418 for the third category, and 0.01613 for the fourth category. This suggests that these networks perform exactly at the average level in the first category, slightly better than average in the second category, significantly better than average in the third category, and moderately better than average in the fourth category.

TABLE 3. Negative Distance from average (PDA)

Negative Distance from Average (NDA)			
0.00000	0.02597	0.00000	0.00000
0.00758	0.12338	0.00000	0.00000
0.00000	0.00000	0.07463	0.06452
0.01732	0.00000	0.34328	0.16129
0.00433	0.06494	0.00000	0.00000
0.00000	0.02597	0.16418	0.01613

Table 3 shows the negative distance from the average The NDA values represent how each type of neural network deviates from the average in each category. A value of 0.00000 indicates that the network's performance is exactly at the average level in that category. Positive values indicate that the network performs better than the average in that category. Negative values indicate that the network performs worse than the average in that category. CNNs perform slightly above average in the second category (0.02597). GNNs perform significantly above average in the second category (0.12338) and moderately above average in the first category (0.00758).RNNs perform moderately above average in the fourth category (0.06452).GANs perform significantly below average in the third category (-0.34328) and moderately below average in the fourth category (-0.16129).HRL performs moderately above average in the second category (0.06494) and slightly above average in the first category (0.00433).Attention Mechanism-based Networks perform significantly above average in the third category (0.16418) and moderately above average in the fourth category (0.01613). These values provide insight into how each type of neural network compares to the average performance across different categories.

TABLE 4. Weighted PDA (SPi)

Weighted PDA				**SPi**
0.00027	0.00000	0.02612	0.00806	0.03445
0.00000	0.00000	0.07090	0.03226	0.10315
0.00216	0.01786	0.00000	0.00000	0.02002
0.00000	0.04221	0.00000	0.00000	0.04221
0.00000	0.00000	0.04851	0.02016	0.06867
0.00487	0.00000	0.00000	0.00000	0.00487

Table 4 shows explanation of weighted PDA. Higher values suggest that the corresponding model has a higher contribution to the detection alarm probability. For example: For CNNs, the highest contribution to the detection alarm probability comes from the third feature (0.02612), followed by the fifth feature (0.03445). For GNNs, the highest contribution comes from the third feature (0.07090), followed by the fifth feature (0.10315). For RNNs, the highest contribution comes from the first feature (0.00216), followed by the second feature (0.01786). SPi: This seems to be another set of weighted values t seems these values might represent the weighted contribution of each model to some measure denoted by SPi. Similarly, higher values indicate a higher contribution of the corresponding model to SPi. For example: For CNNs, the highest contribution to SPi comes from the third feature (0.02612), followed by the fourth feature (0.00806). For GNNs, the highest contribution comes from the fifth feature (0.10315), followed by the third feature (0.07090). For RNNs, the highest contribution comes from the fifth feature (0.02002), followed by the first feature (0.00216).

TABLE 5. Weighted NDA (SNi)

Weighted NDA				SNi
0.00000	0.00649	0.00000	0.00000	0.00649
0.00189	0.03084	0.00000	0.00000	0.03274
0.00000	0.00000	0.01866	0.01613	0.03479
0.00433	0.00000	0.08582	0.04032	0.13047
0.00108	0.01623	0.00000	0.00000	0.01732
0.00000	0.00649	0.04104	0.00403	0.05157

Table 5 shows explanation of weighted NDA. Weighted NDA (Negative Detection Alarm): This could represent the weighted contribution of each model towards the overall negative detection alarm, which might indicate the likelihood of not detecting an anomaly or event of interest. Higher values suggest that the corresponding model has a higher contribution to the negative detection alarm. For example: For CNNs, the highest contribution to the negative detection alarm comes from the second feature (0.00649). For GNNs, the highest contribution comes from the fifth feature (0.03274), followed closely by the second feature (0.03084). For RNNs, the highest contribution comes from the fifth feature (0.03479), followed by the third feature (0.01866). SNi: it seems these values might represent the weighted contribution of each model to some measure denoted by SNi. Similarly, higher values indicate a higher contribution of the corresponding model to SNi. For example: For CNNs, the highest contribution to SNi comes from the second feature (0.00649), followed by the fifth feature (0.00649). For GNNs, the highest contribution comes from the fifth feature (0.03274), followed by the second feature (0.03084). For RNNs, the highest contribution comes from the fifth feature (0.03479), followed by the third feature (0.01613).

TABLE 6. Spi&Sni&ASI&Rank

NSPi	NSPi	ASi	Rank
0.33401	0.95023	0.64212	3
1.00000	0.74908	0.87454	1
0.19410	0.73339	0.46374	4
0.40917	0.00000	0.20459	6
0.66569	0.86728	0.76649	2
0.04721	0.60474	0.32598	5

Table 6 shows the Evolution of structured deep visual models in robot manipulation final result of this paper the Attention Mechanism-based Networks is in 5th rank, Graph Neural Networks (GNNs) is in 1st rank, Hierarchical Reinforcement Learning (HRL) is in 2nd rank, Convolution Neural Networks (CNNs) is in 3rd rank, Generative Adversarial Networks (GANs) is in 6th rank, Recurrent Neural Networks (RNNs) is in 4th rank.

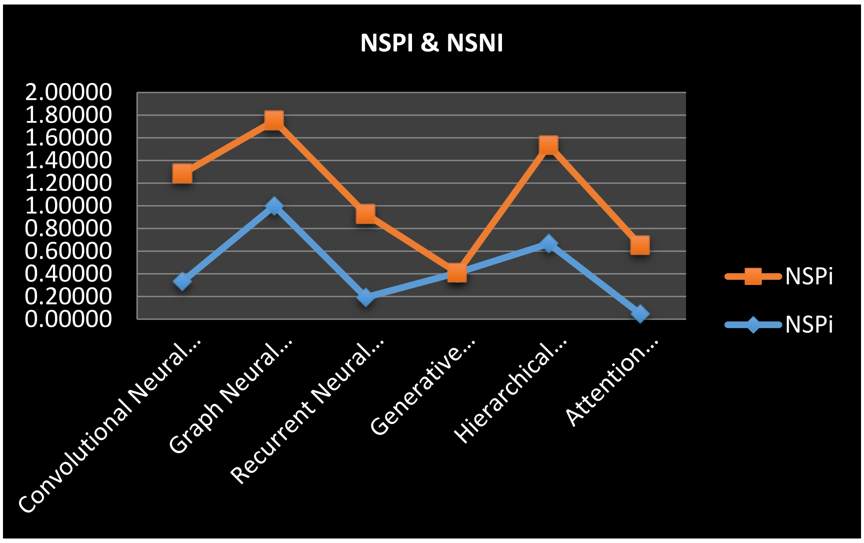


FIGURE 2. Spi&Sni

Figure 2 shows the graphical representation Evolution of structured deep visual models in robot manipulation SPi refers to positive average value and SNi refers to negative average value.

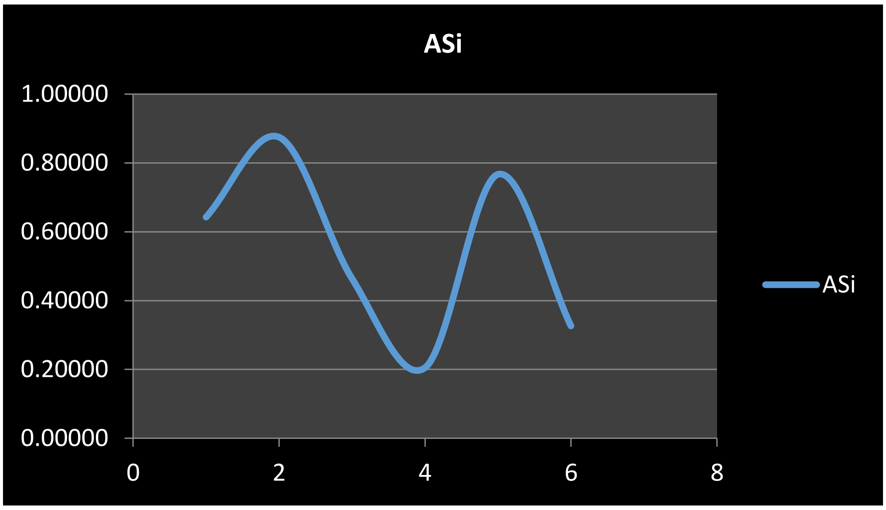


FIGURE 3. ASi

Figure 3 shows the graphical representation of Evolution of structured deep visual models in robot manipulation ASi value. Calculate the average value for positive and negative values. Convolutional Neural Networks (CNNs) is 0.64212; Graph Neural Networks (GNNs) is 0.87454, Recurrent Neural Networks (RNNs) 0.46374, Generative Adversarial Networks (GANs) 0.20459, Hierarchical Reinforcement Learning (HRL) 0.76649, and Attention Mechanism-based Networks 0.32598.

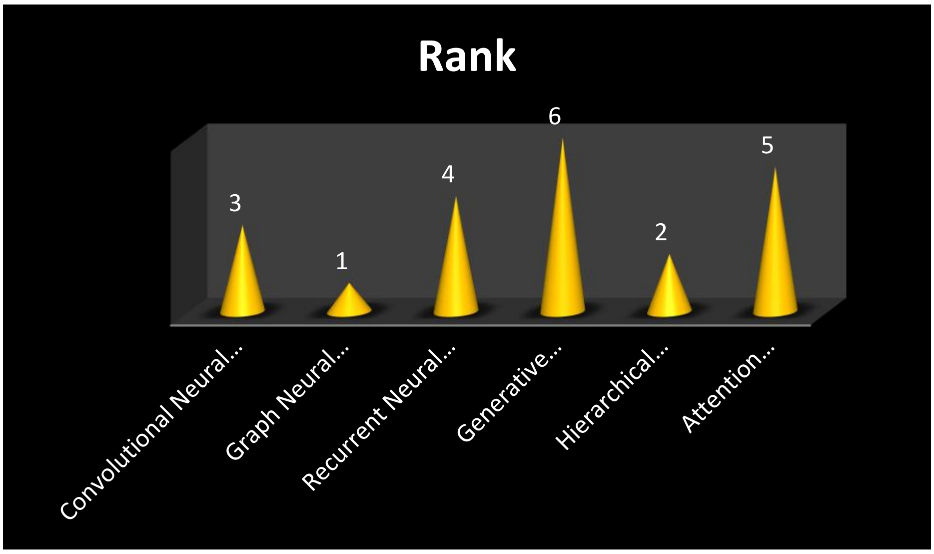


FIGURE.4 Ranks

Figure 4 illustrate graphical representation of Graph Neural Networks (GNNs) is got the first rank where as Generative Adversarial Networks (GANs) is having the lowest rank.

4. CONCLUSION

The evolution of structured deep visual models in robot manipulation signifies a pivotal advancement in robotics research. By integrating deep learning methodologies with structured representations, such as graphs or hierarchical architectures, these models exhibit enhanced interpretability and resilience in intricate manipulation tasks. This evolution has empowered robots to perceive and engage with their surroundings more intelligently, resulting in heightened performance and adaptability across diverse real-world scenarios. Moreover, the symbiosis between structured representations and deep learning fosters efficient knowledge transfer and generalization capabilities, facilitating the seamless application of learned insights to novel tasks. As a result, these sophisticated models not only contribute to the refinement of robotic manipulation techniques but also hold immense promise for revolutionizing various domains, including manufacturing, healthcare, and service industries, by enabling the development of more versatile and autonomous robotic systems capable of tackling complex challenges with precision and efficiency.

REFERENCES

[1].Károly, Artúr István, Péter Galambos, József Kuti, and Imre J. Rudas. "Deep learning in robotics: Survey on model structures and training strategies." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 51, no. 1 (2020): 266-279.

[2].Ribeiro, Eduardo Godinho, Raul de Queiroz Mendes, and Valdir Grassi Jr. "Real-time deep learning approach to visual servo control and grasp detection for autonomous robotic manipulation." *Robotics and Autonomous Systems* 139 (2021): 103757.

[3].Sundaresan, Priya, Jennifer Grannen, Brijen Thananjeyan, Ashwin Balakrishna, Michael Laskey, Kevin Stone, Joseph E. Gonzalez, and Ken Goldberg. "Learning rope manipulation policies using dense object descriptors trained on synthetic depth data." In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9411-9418. IEEE, 2020.

[4].Li, Chih-Hung Gilbert, Long-Ping Zhou, and Yu-Hua Chao. "Self-balancing two-wheeled robot featuring intelligent end-to-end deep visual-steering." *IEEE/ASME Transactions on Mechatronics* 26, no. 5 (2020): 2263-2273.

[5].Haarnoja, Tuomas, Vitchyr Pong, Aurick Zhou, Murtaza Dalal, Pieter Abbeel, and Sergey Levine. "Composable deep reinforcement learning for robotic manipulation." In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 6244-6251. IEEE, 2018.

[6].Wang, Hesheng, Weidong Chen, Xiaojin Yu, Tao Deng, Xiaozhou Wang, and Rolf Pfeifer. "Visual servo control of cable-driven soft robotic manipulator." In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 57-62. IEEE, 2013.

[7].Liu, Rongrong, Florent Nageotte, Philippe Zanne, Michel de Mathelin, and Birgitta Dresp-Langley. "Deep reinforcement learning for the control of robotic manipulation: a focussed mini-review." *Robotics* 10, no. 1 (2021): 22.

[8].Seker, M. Yunus, Ahmet E. Tekden, and Emre Ugur. "Deep effect trajectory prediction in robot manipulation." *Robotics and Autonomous Systems* 119 (2019): 173-184.

[9].Yan, Mengyuan, Yilin Zhu, Ning Jin, and Jeannette Bohg. "Self-supervised learning of state estimation for manipulating deformable linear objects." *IEEE robotics and automation letters* 5, no. 2 (2020): 2372-2379.

[10]. Hirose, Noriaki, Fei Xia, Roberto Martín-Martín, Amir Sadeghian, and Silvio Savarese. "Deep visual mpc-policy learning for navigation." *IEEE Robotics and Automation Letters* 4, no. 4 (2019): 3184-3191.

[11]. Karatop, Buket, Buşra Taşkan, Elanur Adar, and Cemalettin Kubat. "Decision analysis related to the renewable energy investments in Turkey based on a Fuzzy AHP-EDAS-Fuzzy FMEA approach." Computers & Industrial Engineering 151 (2021): 106958.

[12]. Li, Ying-ying, Jian-qiang Wang, and Tie-li Wang. "A linguistic neutrosophic multi-criteria group decision-making approach with EDAS method." Arabian Journal for Science and Engineering 44, no. 3 (2019): 2737-2749.

[13]. Zhan, Jianming, Haibo Jiang, and Yiyu Yao. "Covering-based variable precision fuzzy rough sets with PROMETHEE-EDAS methods." Information Sciences 538 (2020): 314-336.

[14]. Schitea, Dorin, Muhammet Deveci, Mihaela Iordache, Kürşad Bilgili, Ibrahim Zeki Akyurt, and Ioan Iordache. "Hydrogen mobility roll- up site selection using intuitionistic fuzzy sets based WASPAS, COPRAS and EDAS." International Journal of Hydrogen Energy 44, no. 16 (2019): 8585-8600.

[15]. Asante, Dennis, Zheng He, Nana Osae Adjei, and Bismark Asante. "Exploring the barriers to renewable energy adoption utilizing MULTIMOORA-EDAS method." Energy Policy 142 (2020): 111479.

[16]. Wei, Guiwu, Cun Wei, and Yanfeng Guo. "EDAS method for probabilistic linguistic multiple attribute group decision making and their application to green supplier selection." Soft Computing 25, no. 14 (2021): 9045-9053.

[17]. Mishra, Arunodaya Raj, Pratibha Rani, and Kiran Pandey. "Fermatean fuzzy CRITIC-EDAS approach for the selection of sustainable third-party reverse logistics providers using improved generalized score function." Journal of ambient intelligence and humanized computing 13, no. 1 (2022): 295-311.

[18]. Darko, Adjei Peter, and Decui Liang. "Some q-rung orthopair fuzzy Hamacher aggregation operators and their application to multiple attribute group decision making with modified EDAS method." Engineering Applications of Artificial Intelligence 87 (2020): 103259.

[19]. Mishra, Arunodaya Raj, Abbas Mardani, Pratibha Rani, and Edmundas Kazimieras Zavadskas. "A novel EDAS approach on intuitionistic fuzzy set for assessment of health-care waste disposal technology using new parametric divergence measures." Journal of Cleaner Production 272 (2020): 122807.

[20]. Krishankumar, Raghunathan, Dragan Pamucar, Muhammet Deveci, and Kattur Soundarapandian Ravichandran. "Prioritization of zero- carbon measures for sustainable urban mobility using integrated double hierarchy decision framework and EDAS approach." Science of the Total Environment 797 (2021): 149068.

[21]. Kahraman, Cengiz, Mehdi Keshavarz Ghorabaee, Edmundas Kazimieras Zavadskas, Sezi Cevik Onar, Morteza Yazdani, and Basar Oztaysi. "Intuitionistic fuzzy EDAS method: an application to solid waste disposal site selection." Journal of Environmental Engineering and Landscape Management 25, no. 1 (2017): 1-12.

[22]. Peng, Xindong, and Chong Liu. "Algorithms for neutrosophic soft decision making based on EDAS, new similarity measure and level soft set." Journal of Intelligent & Fuzzy Systems 32, no. 1 (2017): 955-968.

[23]. Ghorabaee, Mehdi Keshavarz, Maghsoud Amiri, Edmundas Kazimieras Zavadskas, Zenonas Turskis, and Jurgita Antucheviciene. "A new multi-criteria model based on interval type-2 fuzzy sets and EDAS method for supplier evaluation and order allocation with environmental considerations." Computers & Industrial Engineering 112 (2017): 156-174.Huang, Yuhan, Rui Lin, and Xudong Chen. "An enhancement EDAS method based on prospect theory." Technological and Economic Development of Economy 27, no. 5 (2021): 1019-1038.

